REMARKS

Claims 1-9 were pending in this application. Claim 8 has been amended. Thus, claims 1-9 remain pending, with claims 1, 5, 8 and 9 being independent claims.

Rejections under 35 U.S.C. §102

Claims 1-5 and 8-9 stand rejected under 35 U.S.C. §102 (e) as purportedly being anticipated by U.S. Patent No. 5, 995, 736 to Aleksic et al. (hereinafter Aleksic). Applicant respectfully traverses this rejection.

A. Discussion of Aleksic

Aleksic is directed to a method and system for automatically modeling registers for integrated circuit design. Specifically, Aleksic is directed to developing a software model to simulate an Application Specific Integrated Circuit (ASIC) that is not yet implemented in hardware. In general, the system of Aleksic automatically generates behavioral model register code and hardware design simulation code from a common input or source (col. 3, lines 53-58). The common source is a text file, in which the registers to be modeled are defined (col. 3, lines 59-66). The system uses the common input source to automatically generate models of the registers in different languages (col. 3, line 66-col. 4, line 3).

Referring to Figure 2, the system for automatically modeling registers includes a model register generator 36 and common register description source data 34, which is a text file (col. 5, lines 19-22, lines 34-36). An example of register description source data 34 is illustrated in Figure 6A (col. 9, lines 7-9). As previously mentioned, the register description source data 34 defines the registers to be modeled (col. 3, lines 59-66). In operation, model register generator 36 accesses the source data 34 and generates several outputs: hardware design simulation code 38, behavioral model register code 40, application interface layer code 42, automatic register tests 44, automatic integrated circuit documentation 46, automatic software header information 48, and connection templates 49 (col. 5, lines 37-39).

B. Claim 1 Distinguishes Over Aleksic

Claim 1 is directed to a method of operating a computer system to design an application specific processor (ASP). The method comprises defining a set of peripherals for the ASP which are responsive to stimuli and which communicate with a processor. The method further comprises generating for each peripheral an input file which defines the functional attributes of that peripheral in a high level language with an input data structure. The method further comprises entering the input file into the computer system and operating a modelling tool loaded on the computer system to generate from the input file a register definition file by allocating specific elements of the input data structure to predefined sectors of a register definition table. The method further comprises using the register definition file to create in silicon the registers of the ASP.

Aleksic fails to teach or suggest several limitations of claim 1. For example, Aleksic fails to teach or suggest the limitation of generating for each peripheral an **input file** which defines the functional attributes of that peripheral **in a high level language** with an input data structure. Although the Office Action does not explicitly cite which feature of Aleksic corresponds to the claimed input file, it appears that the Office Action correlates the common register description source data 34 of Aleksic to the claimed input file. However, such a correlation is inaccurate. The common register description source date 34 is in a text format, not a high level language, as claimed. Therefore, the common register description source data 34 of Aleksic does not correspond to the claimed input file.

Furthermore, Aleksic fails to teach or suggest entering the input file into the computer system and operating **a modelling tool** loaded on the computer system **to generate** from the input file **a register definition file by allocating specific elements of the input data structure to predefined sectors of a register definition table**, as claimed.

First, Aleksic does not teach or suggest the claimed modeling tool. Although the Office Action does not explicitly cite which feature of Aleksic corresponds to the claimed modeling tool, it appears that the Office Action correlates the model register generator 36 of Aleksic to the claimed modeling tool. However, the model register generator 36 of Aleksic reads in and operates on the common register description source date 34, which is in a text format, not a high

level language. Aleksic does not teach or suggest that the model register generator 36 operates on an input file in a high level language. The model register generator 36 of Aleksic also does not generate a register definition file. Rather, the model register generator 36 accesses a common input source in which the registers are already defined and generates model code in different languages (col. 3, lines 61-col. 4, line 3). Therefore, the model register generator 36 of Aleksic does not correspond to the claimed modeling tool.

Secondly, Aleksic does not teach or suggest allocating specific elements of the input data structure to predefined sectors of a register definition table. In fact, no such table is taught or suggested by Aleksic. In this regard, it is unclear from paragraph 11.3, on page 11 of the Office Action, which feature of Aleksic the Examiner believes corresponds to the claimed register definition table. If the Examiner is to maintain the position that Aleksic does disclose such a feature, Applicant respectfully requests that the Examiner explicitly cite the feature of Aleksic which corresponds to the claimed register definition table.

Since Aleksic fails to teach or suggest at least the above-referenced limitations of claim 1, Applicant respectfully requests that the rejection of claim 1 under 35 U.S.C. §102(e) be withdrawn.

Claims 2-4 depend from claim 1 and are allowable for at least the same reasons. Accordingly, Applicant respectfully requests that the rejection of claim 2-4 under 35 U.S.C. §102(e) be withdrawn.

C. Claim 5 Distinguishes Over Aleksic

Claim 5 is directed to a computer system which comprises a processor and a memory, the memory holding a program representing a modelling tool for use in designing an application specific processor (ASP). The computer system comprises an input means for receiving a plurality of input files, each input file defining the functional attributes of a peripheral for the ASP in a high level language within an input data structure. The processor is operable to execute the program representing the modelling tool to generate from the input file a register definition file by allocating specific elements of the input data structure to predefined sectors of a register

definition table. The computer system further comprises an output means for outputting the register definition file in a manner which is usable to create in silicon the registers of the ASP.

Aleksic fails to teach or suggest several limitations of claim 5. For example, as described in connection with claim 1, Aleksic fails to teach or suggest the claimed **input files** defining the functional attributes of a peripheral for the ASP **in a high level language**. Furthermore, Aleksic fails to teach or suggest the claimed **modeling tool to generate** from the input file **a register definition file**. Aleksic also fails to teach or suggest allocating specific elements of the input data structure to predefined sectors of a register definition table.

Since Aleksic fails to teach or suggest at least the above-referenced limitations of claim 5, Applicant respectfully requests that the rejection of claim 5 under 35 U.S.C. §102(e) be withdrawn.

Claims 6-7 depend from claim 5 and are allowable for at least the same reasons. Therefore, Applicant respectfully requests that the rejection of claims 6-7 under 35 U.S.C. §103(a) be withdrawn.

D. Claim 8 Distinguishes Over Aleksic

Claim 8 is directed to a computer program product stored on a computer readable medium. The computer program product comprises software code portions operable when executed by a computer to read an input file which defines in an input data structure the functional attributes of a peripheral for an application specific processor in a high level language, and to generate from that input file a register definition file. The software code portions include a code portion for allocating specific elements of the input data structure to predefined sectors of a register definition table for each of a plurality of registers.

Aleksic fails to teach or suggest several limitations of claim 8. For example, as discussed in connection with claim 1, Aleksic fails to teach or suggest the claimed input file which defines in an input data structure the functional attributes of a peripheral in a high level language. Aleksic also fails to teach or suggest software code portions operable to read an input file and to generate from that input field a register definition file. Aleksic also fails to teach or suggest allocating specific elements of the input data structure to predefined sectors of a register

definition table, as claimed. Accordingly, Applicant respectfully requests that the rejection of claim 8 under 35 U.S.C. §102(e) be withdrawn.

E. Claim 9 Distinguishes Over Aleksic

Claim 9 is directed to a register definition file stored on a computer readable medium and comprising a plurality of register definition tables. Each table includes at least predefined sectors for the bit location within a register of an element, the name of the element, the function of the element and the functional status of the element. Each table further includes the word location of the register within a memory map, wherein when the computer readable medium is loaded into a computer for simulating an Application Specific Processor, said register definition tables are accessed based on said word location for simulation of said registers.

Contrary to the assertion on page 6 of the Office Action, Aleksic does not teach or suggest the limitations of claim 9. Specifically, the Office Action refers to Figure 7 of Aleksic, which illustrates the modeling of an integrated circuit based on behavioral model code templates 60, which are part of the register templates in the register generator 36 (col. 11, lines 51-53, Figure 3). Therefore, Figure 7 does not illustrate a register definition file. Furthermore, the portions of Aleksic to which the Office Action refers do not teach or suggest that the register definition tables are accessed based on said word location for simulation of said registers, as claimed. Accordingly, Applicant respectfully requests that the rejection of claim 9 under 35 U.S.C. §102(e) be withdrawn.

CONCLUSION

In view of the foregoing remarks, this application should now be in condition for allowance. A notice to this effect is respectfully requested. If the Examiner believes that the application is not in condition for allowance, the Examiner is requested to call the Applicant's attorney at the telephone number listed below.

If this response is not considered timely filed and if a request for an extension of time is otherwise absent, Applicant hereby requests any necessary extension of time. If there is a fee occasioned by this response, including an extension fee, that is not covered by an enclosed check, please charge any deficiency to Deposit Account No. 23/2825.

Respectfully submitted,

Gajinder Singh PANESAR, Applicant

By: _____

James H. Morris, Reg. No.: 34,681 Wolf,
Greenfield & Sacks, P.C.
600 Atlantic Avenue
Boston, Massachusetts 02210-2211
Telephone: (617) 720-3500